Joint University Program
for
Air Transportation Research

Laboratory
for
Information
and Decision
Systems

Natural Language Interface for Air Traffic Control

Emily M. Craparo

Laboratory for Information and Decision Systems (LIDS)

Massachusetts Institute of Technology

Joint University Program for Air Transportation Research

Quarterly Review - June 19, 2003

Summary of Contributions

We have developed a framework for a natural language interface to an unmanned aerial vehicle:

• Parser handles over 100 ATC commands (text input)

• Discourse manager interacts with the user and interprets commands at a higher level than individual sentences

• Simulation environment is modeled on an actual airport (Laurence G. Hanscom Field in Bedford, MA)

• Graphical user interface allows real time interaction with aircraft

Outline

- Background and Motivation
- Suitability of ATC for NLP
- Our System:
  - Overall Structure
  - Preprocessor
  - Sentence Parser
  - Discourse Manager
  - Airport/Airplane Modeling
- Demos
- Future Work
- Acknowledgements
- References
- Questions

- Increased complexity in ATC has lead to **efforts to automate the process**. Automation in ATC increases efficiency, but it also raises questions about **adequate human control over the automated systems**. As a consequence, significant amount of research has been focused on the technology that **builds on the human strengths and compensates for human vulnerabilities** [Wicken 98].

- *Churcher et al.* intended to use speech recognition technology to automatically **transcribe certain, essential parts of transmissions between the air traffic control (ATC) and airborne pilots** [Churcher 96].

  - 30% accuracy was achieved with an off the shelf IBM speech recognizer

  - 70% accuracy was achieved with enhanced contextual information
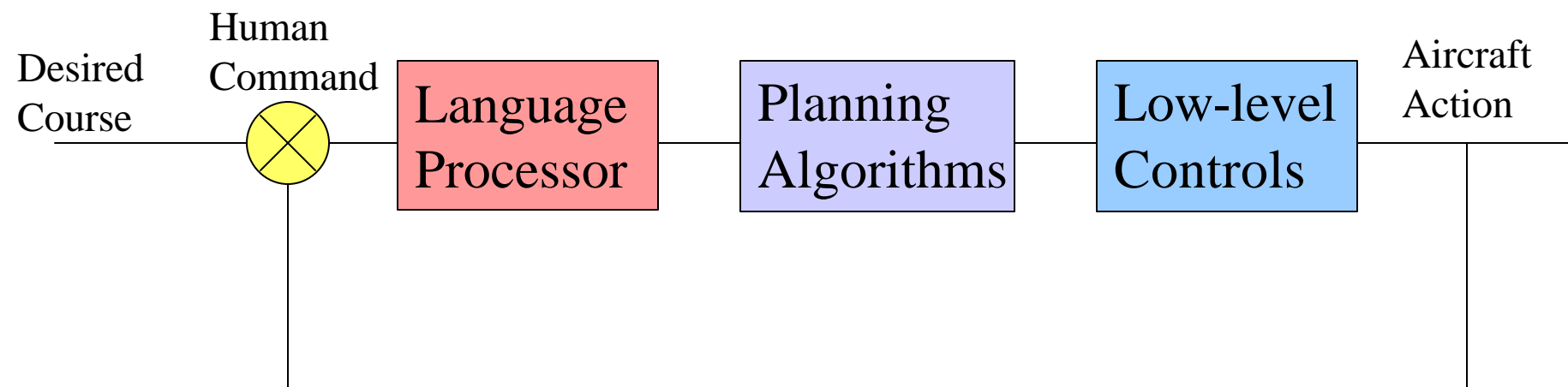
Motivation – UAV Control

- Current UAV control schemes
  - Are low-level and relatively unintuitive
  - Require a high degree of human supervision
  - Are poorly suited to multiple UAV-operations
- Natural language control
  - Is high-level and intuitive
  - Requires relatively little human supervision
  - Is useful for multiple UAVs

NLP in UAV Control

Desired Course

Human Command

Language Processor

Planning Algorithms
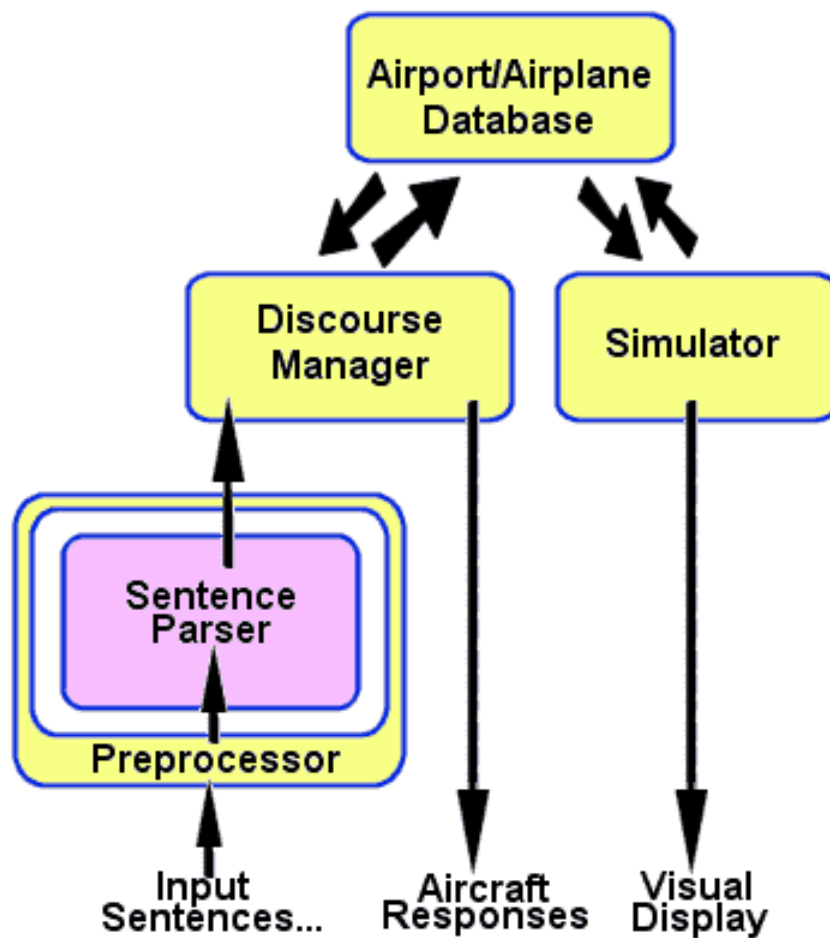
Low-level Controls

Aircraft Action

Suitability of ATC for NLP

- Limited vocabulary and domain of discourse

- Compact and standard syntactical structure aids in parsing efficiency and accuracy

- Task-oriented nature of air traffic control makes intentional inference possible

Structure of the Interface

Airport/Airplane Database

Discourse Manager

Simulator

Sentence Parser

Preprocessor

Input Sentences...

Aircraft Responses

Visual Display

Structure of the Interface

- Modular: allows multiple collaborators to work together

- Aids in troubleshooting

- Makes further optimization/future replacement of modules easier

Preprocessor/C++ Wrapper

Converts input to proper form for Lisp parser, and converts output to proper form for C++ modules:

- Converts uppercase letters to lowercase

- Converts punctuation to spaces

- Inserts $ before numbers

- Expands contractions

- Runs Lisp parser in the background

The Sentence Parser

- Based on the Earley context free parser
- Recognizes sentence structures derived from a corpus of actual air traffic control exchanges at Boston's Logan Airport and Laurence G. Hanscom Field in Bedford, MA
- Converts sentences to verb templates

The Sentence Parser

```
; Replace "clear to <vp>" with "you <vp>"
(add-rule-sem '(s ==> clear to vp) '(lambda (a1 a2 a3) (funcall a3
'you)))

;takeoff on? <rw>
(add-rule-sem '(v+args ==> takeoff on? rw) '(lambda (a1 a2 a3) `(lambda
(subj) (print-template `(takeoff :on ,',a3 :agent ,subj)))))

(add-rule-sem '(rw ==> runway some-number) '(lambda (a b) `(runway :num
,b)))

(add-rule-sem '(rw ==> some-number) '(lambda (a) `(runway :num ,a)))


(add-number 'twoniner '$29 '29)
```

sentence: "clear to takeoff runway twoniner"
[OUTPUT] (takeoff :on (runway :num 29) :agent you)

The Sentence Parser

```
sentence: "clear to land"
[OUTPUT] (land :agent you)

sentence: "clear to land runway five"
[OUTPUT] (land :on (runway :num 5) :agent you)

sentence: "clear to land runway five left"
[OUTPUT] (land :left (runway :num 5) :agent you)

sentence: "clear for landing runway twoniner"
[OUTPUT] (land :on (runway :num 29) :agent you)
```

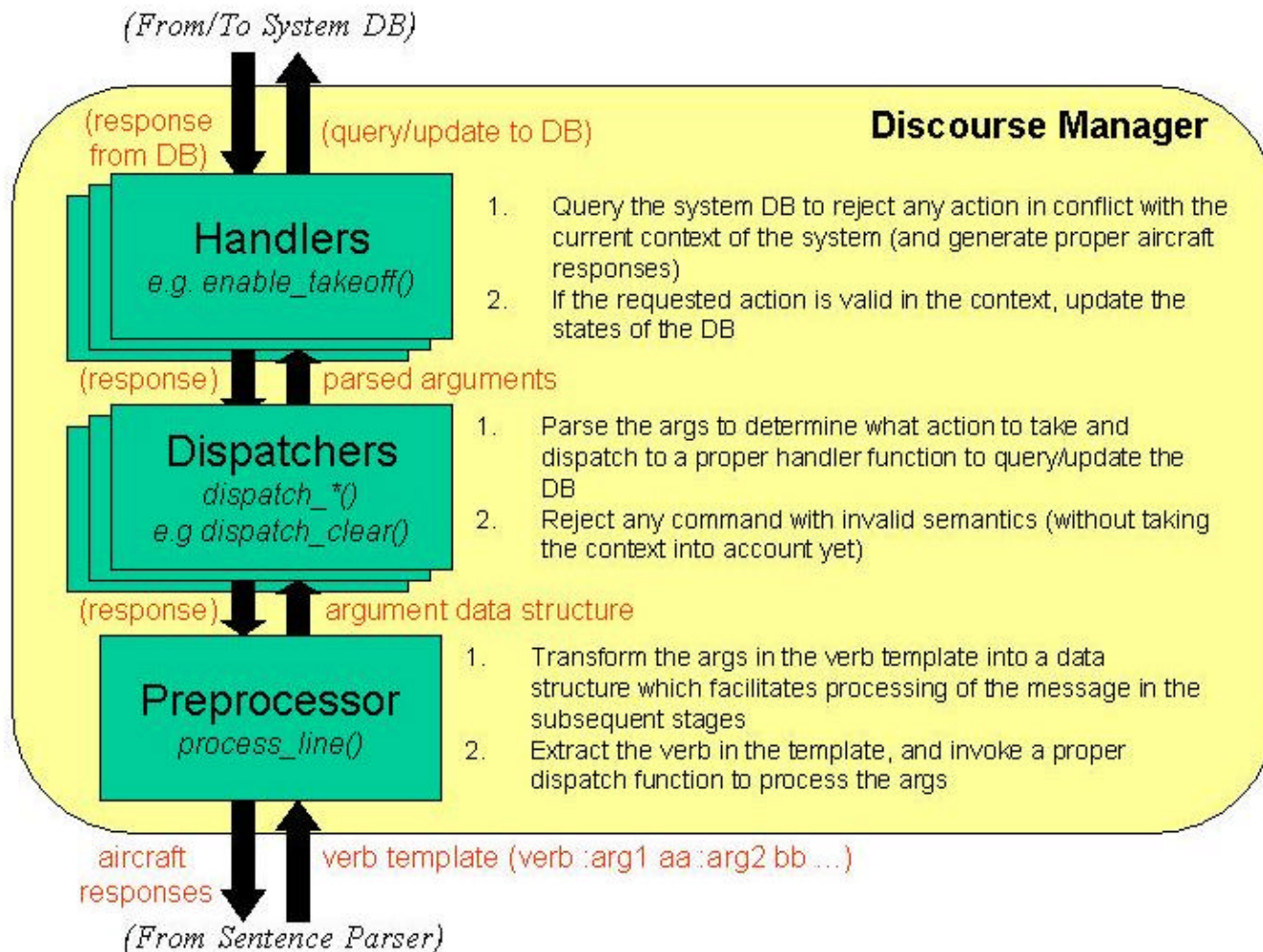**May overgenerate (parse nonsensical sentences)**

The Discourse Manager

- Performs semantic interpretation on verb templates generated by the parser

- Resolves any ambiguities in commands by referring to the current system state and dominant intention

- Analyzes consecutive commands for consistency

- Updates the database or generates a response to the user depending on the request and the context of the request

The Discourse Manager

(From/To System DB)

Discourse Manager

(response from DB)   (query/update to DB)

Handlers
e.g. enable_takeoff()

1. Query the system DB to reject any action in conflict with the current context of the system (and generate proper aircraft responses)
2. If the requested action is valid in the context, update the states of the DB

(response)   parsed arguments

Dispatchers
dispatch_*()
e.g dispatch_clear()

1. Parse the args to determine what action to take and dispatch to a proper handler function to query/update the DB
2. Reject any command with invalid semantics (without taking the context into account yet)

(response)   argument data structure

Preprocessor
process_line()

1. Transform the args in the verb template into a data structure which facilitates processing of the message in the subsequent stages
2. Extract the verb in the template, and invoke a proper dispatch function to process the args

aircraft responses   verb template (verb ::arg1 aa ::arg2 bb ...)

(From Sentence Parser)

Airport/Aircraft Database

- Contains states of the airport and all aircraft being controlled

- Consulted and updated by the discourse manager

- Airport model based on Laurence G. Hanscom field; modeled as a set of points and an adjacency matrix

- Airport state includes takeoff and landing priority queues

- Airplanes modeled as C++ objects with states such as position, altitude, and speed

- Airplanes can query one another's positions to avoid collisions

- Airplane states include status flags to indicate current activity

- Previous airplane state is maintained in order to facilitate command cancellation and implicit references
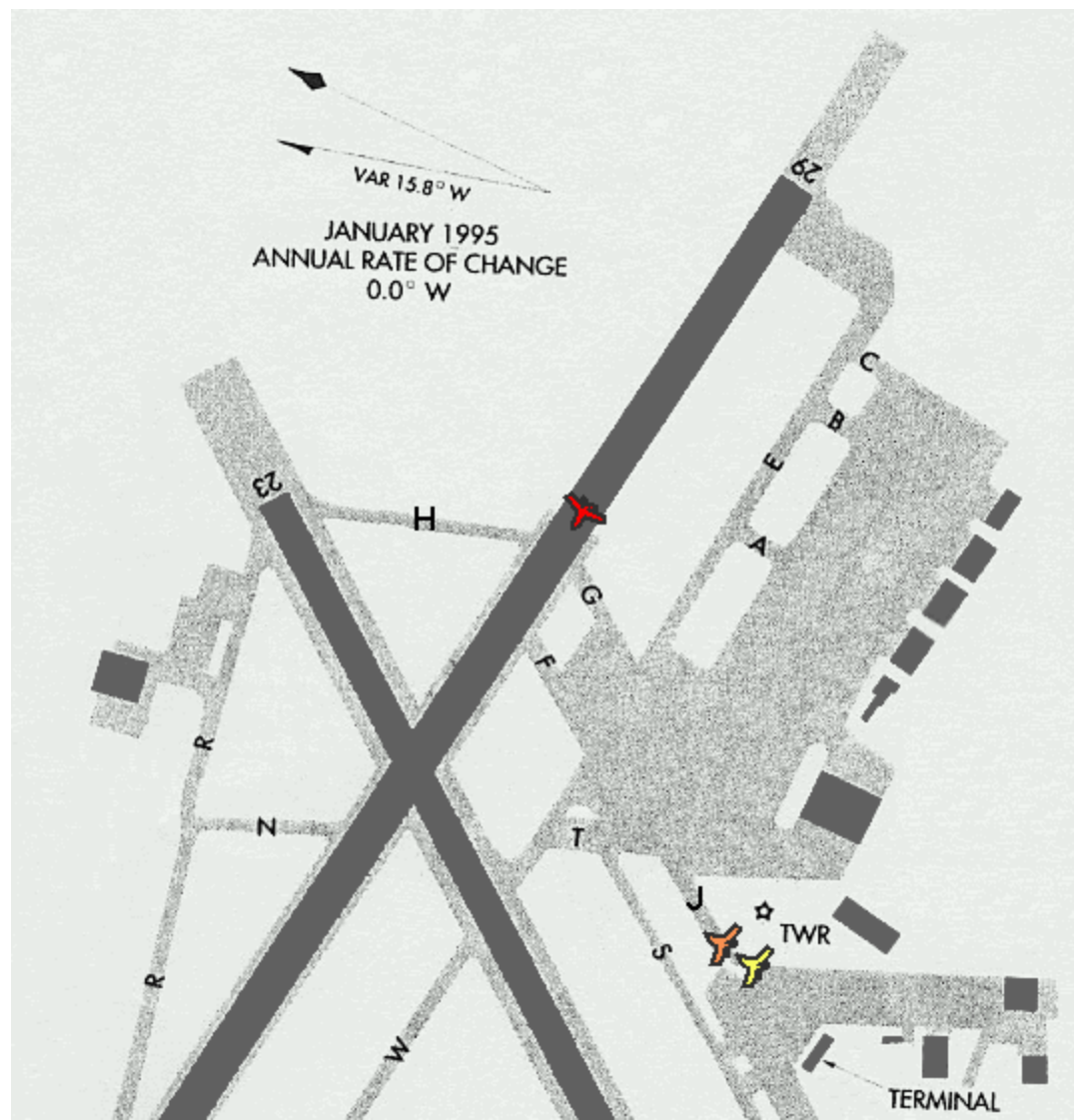
VAR 15.8° W

JANUARY 1995
ANNUAL RATE OF CHANGE
0.0° W

Alaska to controller: Ready to taxi.

Boeing to controller: Ready to taxi.

Cessna to controller: Ready to taxi.

Alaska> Alaska, clear to taxi to runway 5.
Roger.

Horizon to controller: Inbound.

United to controller: Inbound.

Alaska> Cancel that.
Roger. Awaiting a new destination.

Alaska> Taxi to twoniner via echo.
Roger.

Alaska to controller: Doing runup...

Alaska to controller: Run up completed. Ready for departure.

Alaska> Alaska, hold short.
Roger.

Alaska> Alaska, taxi into position and hold.
Roger.

Alaska to controller: Position and hold. Ready for departure.

Alaska> Clear for takeoff.
Roger.

Alaska> █

Demo 2 - Landing
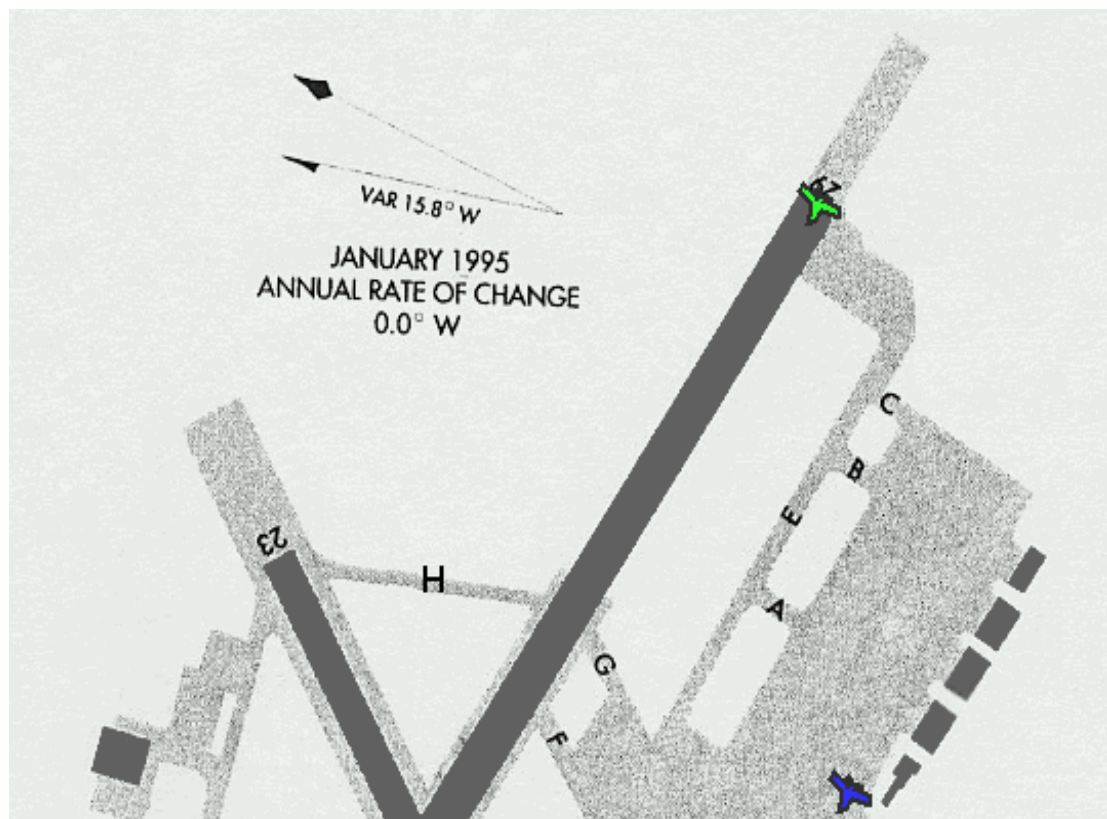
The third scenario involves 2 aircraft (Alaska and Boeing) who both need to cross the same intersection. The controller had previously told Alaska (shown here in red) to hold for Boeing (meaning Alaska has a lower priority).
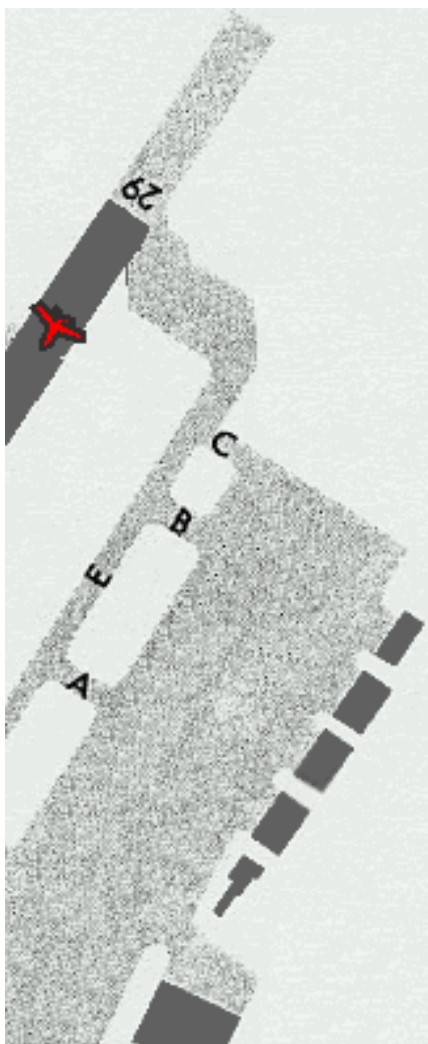
Controller: Alaska, hold for the Boeing.
Alaska: Roger.

If they should both reach the intersection at about the same time, Boeing (shown here in orange) will wait until Alaska passes.
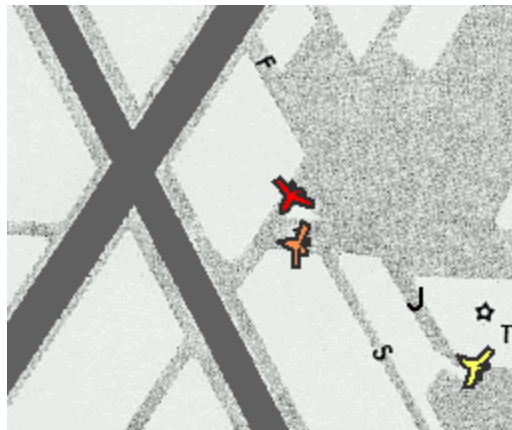
Demo 4 - Controller Error

Laboratory for Information and Decision Systems

Alaska to controller: Ready to taxi.

Boeing to controller: Ready to taxi.

Cessna to controller: Ready to taxi.

Alaska> Clear to runway twoniner.
Roger.

Horizon to controller: Inbound.

United to controller: Inbound.

Alaska> Taxi to runway ten via echo.
I do not think there is such a runway.

Alaska> Taxi to runway twoniner via echo.
Roger.

Alaska to controller: Doing runup...

Alaska to controller: Run up completed. Ready for departure.

Alaska> Position and hold.
Roger.

Alaska to controller: Position and hold. Ready for departure.

Alaska> Clear for takeoff.
Roger.

Future Work

There is a great potential for improvements and expansions in this area, including

• Integration of optimal path planning algorithms for aircraft in flight

• Multiple-aircraft interface

• Cooperative control strategies for multiple-aircraft operations

• More sophisticated discourse manager - time-sensitive discourse, improved intentional inference

• Hardware implementation

Acknowledgements

References

[Carberry 90]: S. Carberry: *Plan Recognition in Natural Language Dialog*, MIT press, 1990.

[Churcher 96]: Gavin E. Churcher, Eric S. Atwall, and Clive Souter: "Dialogues in Air Traffic Control", *Proceedings of the 11th Twente Workshop on Language Technology*, 1996.

[FAA 2002]: *"Airport/Facility Directory, Northeastern U.S."*, National Aeronautical Charting Office, Federal Aviation Administration, U.S. Department of Transportation, 2002.

[Fed 2000]: *"2002 Federal Aviation Regulations and Aeronautical Information Manual"*, Federal Aviation Administration, U.S. Department of Transportation.

[Grosz 77]: B. Grosz: "The Representation and Use of Focus in Dialogue Understanding", Ph.D. thesis, University of California, Berkeley, 1977.

[Grosz 86]: Barbara Grosz and Candance L. Sidner: "Attention, Intentions, and the Structure of Discourse", *Computational Linguistics*, 12(3), pages 175-204, 1986.

[Hobbs 79]: J.R. Hobbs: "Coherence and Coreference", *Cognitive Science*, 3, pages 67-90.

References

[JS 2001]: *"Private Pilot Manual"* (copyright 2001), Jeppenson Sanderson, Inc., Englewood, CO, 80112-5498.

[Jurafsky]: D. Jurafsky and J.H. Martin: *Speech and Language Processing*, Prentice Hall, 2000.

[Litman and Allen 87]: D.J. Litman and J.F. Allen: "A Plan Recognition Model for Subdialogues in Conversation", *Cognitive Science*, 11, pages 163-200, 1987.

[Passonneau and Litman 93]: R. Passonneau and D.J. Litman: "Intention-based Segmentation: Human Reliability and Correlation with Linguistic Cues", *ACL 93*, Columbus, Ohio.

[Wickens 98]: Christopher D. Wickens, Anne S. Mavor, Raja Parasuraman, and James P. McGee: "The Future of Air Traffic Control: Human Operators and Automation", *Panel on Human Factors in Air Traffic Control Automation*, National Research Council, 1998.

[Yangarber 2000]: Roman Yangarber, Ralph Grishman, Pasi Tapanainen, and Silja Huttunen: "Automatic Acquisition of Domain Knowledge for Information Extraction", *Proceedings of the 18th International Conference on Computational Linguistics*, 2000.

Questions

?

Valid Sentences

```
sentence: "altimeter six"
[OUTPUT] (altimeter :value (6) :agent you)

sentence: "cancel that"
[OUTPUT] (undo :agent you)

sentence: "change of plans"
[OUTPUT] (undo :agent you)

sentence: "change speed to six"
[OUTPUT] (setspeed :goal (6) :agent you)

sentence: "change speed to six knots"
[OUTPUT] (setspeed :goal (6) :agent you)

sentence: "checkin at five o^clock"
[OUTPUT] (checkin :time (timevalue :at 5) :agent you)

sentence: "checkin in five minutes"
[OUTPUT] (checkin :time (timevalue :in 5) :agent you)

sentence: "clear for takeoff"
[OUTPUT] (enable-takeoff :agent you)

sentence: "clear the runway"
[OUTPUT] (leave :goal (runway) :agent you)
```

Valid Sentences

Laboratory
for
Information
and Decision
Systems

```
sentence: "clear the taxiway"
[OUTPUT] (leave :goal (taxiway) :agent you)

sentence: "clear to land"
[OUTPUT] (land :agent you)

sentence: "clear to land runway five"
[OUTPUT] (land :on (runway :num 5) :agent you)

sentence: "clear to land runway five left"
[OUTPUT] (land :left (runway :num 5) :agent you)

sentence: "clear to takeoff runway twoniner"
[OUTPUT] (takeoff :on (runway :num 29) :agent you)

sentence: "clear for landing runway twoniner"
[OUTPUT] (land :on (runway :num 29) :agent you)

sentence: "clear to runway five"
[OUTPUT] (go :on (runway :num 5) :agent you)

sentence: "clear to taxi to two"
[OUTPUT] (go :on (runway :num 2) :agent you)

sentence: "cleared direct to runway five"
[OUTPUT] (go :on (runway :num 5) :agent you)

sentence: "cleared to runway five"
[OUTPUT] (go :on (runway :num 5) :agent you)
```

```
sentence: "climb and maintain three"
[OUTPUT] (climb-maintain :alt (3) :agent you)

sentence: "climb and maintain three feet"
[OUTPUT] (climb-maintain :alt (3) :agent you)

sentence: "contact departure"
[OUTPUT] (setfreq :goal (speaker :name departure) :agent you)

sentence: "contact departure at three"
[OUTPUT] (setfreq :goal (speaker :name departure :freq 3) :agent you)

sentence: "contact ground"
[OUTPUT] (setfreq :goal (speaker :name ground) :agent you)

sentence: "contact ground on five"
[OUTPUT] (setfreq :goal (speaker :name ground :freq 5) :agent you)

sentence: "contact ramp"
[OUTPUT] (setfreq :goal (speaker :name ramp) :agent you)

sentence: "contact ramp on ten"
[OUTPUT] (setfreq :goal (speaker :name ramp :freq 10) :agent you)

sentence: "contact three"
[OUTPUT] (setfreq :goal (speaker :freq 3) :agent you)

sentence: "contact tower"
[OUTPUT] (setfreq :goal (speaker :name tower) :agent you)
```

Valid Sentences

```
sentence: "contact tower on eight"
[OUTPUT] (setfreq :goal (speaker :name tower :freq 8) :agent you)

sentence: "continue on tango"
[OUTPUT] (go :on (taxiway :num tango) :agent you)

sentence: "continue on zulu until runway two"
[OUTPUT] (go :on (taxiway :num zulu) :until (runway :num 2) :agent you)

sentence: "cross runway three at zulu"
[OUTPUT] (cross :road (runway :num 3) :at (taxiway :num zulu) :agent you)

sentence: "cross runway two"
[OUTPUT] (cross :road (runway :num 2) :agent you)

sentence: "cross two"
[OUTPUT] (cross :road (runway :num 2) :agent you)

sentence: "cross two at five knots"
[OUTPUT] (cross :fix 2 :at-speed 5 :agent you)

sentence: "cross two at three feet"
[OUTPUT] (cross :fix 2 :at-altitude 3 :agent you)

sentence: "descend and maintain five"
[OUTPUT] (descend-maintain :alt (5) :agent you)

sentence: "descend and maintain five feet"
[OUTPUT] (descend-maintain :alt (5) :agent you)
```

Valid Sentences

```
sentence: "exit on taxiway tango"
[OUTPUT] (go :on (taxiway :num tango) :agent you)

sentence: "exit the ramp"
[OUTPUT] (leave :goal ramp :agent you)

sentence: "exit the ramp and follow alaska"
[OUTPUT] (leave :goal ramp :agent you)
[OUTPUT] (behind :goal (alaska) :agent you)

sentence: "exit the ramp behind continental"
[OUTPUT] (leave :goal ramp :agent you)
[OUTPUT] (behind :goal (continental) :agent you)

sentence: "expect boeing on tango"
[OUTPUT] (expect :plane (boeing) :on (taxiway :num tango) :agent you)

sentence: "expect three feet five minutes after departure"
[OUTPUT] (lock-altitude :alt 3 :time 5 :agent you)

sentence: "expect traffic on november"
[OUTPUT] (expect :on (taxiway :num november) :agent you)

sentence: "fall in behind the alaska"
[OUTPUT] (behind :goal (alaska) :agent you)

sentence: "follow a continental that is behind a boeing"
[OUTPUT] (behind :goal (continental :behind boeing) :agent you)
```

Valid Sentences

```
sentence: "follow in behind that alaska"
[OUTPUT] (behind :goal (alaska) :agent you)

sentence: "follow that boeing"
[OUTPUT] (behind :goal (boeing) :agent you)

sentence: "follow that boeing ahead to the runway"
[OUTPUT] (behind :goal (boeing) :agent you)
[OUTPUT] (go :on (runway) :agent you)

sentence: "follow that continental directly ahead of you"
[OUTPUT] (behind :goal (continental :ahead you) :agent you)

sentence: "follow the boeing from your left"
[OUTPUT] (behind :goal (boeing :left you) :agent you)

sentence: "get behind the continental"
[OUTPUT] (behind :goal (continental) :agent you)

sentence: "give way to boeing"
[OUTPUT] (behind :goal (boeing) :agent you)

sentence: "give way to the boeing"
[OUTPUT] (behind :goal (boeing) :agent you)

sentence: "go november"
[OUTPUT] (go :on (taxiway :num november) :agent you)
```

```
sentence: "go straight down tango"
[OUTPUT] (go :on (taxiway :num tango) :agent you)


sentence: "hold five"
[OUTPUT] (hold-heading :heading (heading :to 5) :agent you)


sentence: "hold for the continental"
[OUTPUT] (behind :goal (continental) :agent you)


sentence: "hold heading five"
[OUTPUT] (hold-heading :heading (heading :to 5) :agent you)


sentence: "hold short"
[OUTPUT] (hold-short :agent you)


sentence: "hold short of runway three on tango"
[OUTPUT] (hold-short :of (runway :num 3) :on (taxiway :num tango) :agent you)


sentence: "hold short of taxiway zulu"
[OUTPUT] (hold-short :of (taxiway :num zulu) :agent you)


sentence: "hold short of zulu for spacing"
[OUTPUT] (hold-short :of (taxiway :num zulu) :agent you)


sentence: "hold short of zulu for the continental"
[OUTPUT] (hold-short :of (taxiway :num zulu) :for (continental) :agent you)
```

```
sentence: "intercept five"
[OUTPUT] (intercept :patient 5 :agent you)

sentence: "let the boeing turn in front of you"
[OUTPUT] (behind :goal (boeing) :agent you)

sentence: "maintain four feet"
[OUTPUT] (maintain :alt (4) :agent you)

sentence: "maintain four feet at departure"
[OUTPUT] (maintain :alt (4) :when on-departure :agent you)

sentence: "maintain heading of four"
[OUTPUT] (maintain :heading 4 :agent you)

sentence: "maintain this frequency"
[OUTPUT] (nop :agent you)

sentence: "monitor four"
[OUTPUT] (setfreq :goal (speaker :freq 4) :agent you)

sentence: "monitor ground"
[OUTPUT] (setfreq :goal (speaker :name ground) :agent you)

sentence: "monitor ramp"
[OUTPUT] (setfreq :goal (speaker :name ramp) :agent you)

sentence: "monitor tower"
[OUTPUT] (setfreq :goal (speaker :name tower) :agent you)
```

```
sentence: "move ahead"
[OUTPUT] (move :agent you)

sentence: "move ahead before the boeing"
[OUTPUT] (move :agent you)
[OUTPUT] (ahead :goal (boeing) :agent you)

sentence: "move as soon as you can"
[OUTPUT] (move :agent you)

sentence: "on departure maintain five"
[OUTPUT] (maintain :alt (5) :when on-departure :agent you)

sentence: "on departure maintain five feet"
[OUTPUT] (maintain :alt (5) :when on-departure :agent you)

sentence: "position and hold"
[OUTPUT] (position :agent you)
[OUTPUT] (hold :agent you)

sentence: "remain on ten"
[OUTPUT] (setfreq :goal (speaker :freq 10) :agent you)

sentence: "remain this frequency"
[OUTPUT] (nop :agent you)

sentence: "right in front of the continental"
[OUTPUT] (move :agent you)
[OUTPUT] (ahead :goal (continental) :agent you)
```

```
sentence: "straight ahead on zulu"
[OUTPUT] (go :on (taxiway :num zulu) :agent you)

sentence: "straight on to runway three"
[OUTPUT] (go :on (runway :num 3) :agent you)

sentence: "straight on to zulu"
[OUTPUT] (go :on (taxiway :num zulu) :agent you)

sentence: "taxi ahead"
[OUTPUT] (move :agent you)

sentence: "taxi ahead on zulu"
[OUTPUT] (go :on (taxiway :num zulu) :agent you)

sentence: "taxi into position and hold"
[OUTPUT] (position :agent you)
[OUTPUT] (hold :agent you)

sentence: "taxi quebec"
[OUTPUT] (go :on (taxiway :num quebec) :agent you)

sentence: "taxi to position and hold"
[OUTPUT] (position :agent you)
[OUTPUT] (hold :agent you)

sentence: "taxi to runway three via zulu"
[OUTPUT] (go :on (runway :num 3) :via (taxiway :num zulu) :agent you)
```

```
sentence: "taxi via taxiway tango"
[OUTPUT] (go :via (taxiway :num tango) :agent you)

sentence: "the boeing is going in front of you"
[OUTPUT] (behind :goal (boeing) :agent you)

sentence: "then runway three"
[OUTPUT] (go :on (runway :num 3) :agent you)

sentence: "then zulu"
[OUTPUT] (go :on (taxiway :num zulu) :agent you)

sentence: "turn heading three"
[OUTPUT] (turn :heading (heading :to 3) :agent you)

sentence: "turn left five degrees"
[OUTPUT] (turn :heading (heading :left 5) :agent you)

sentence: "turn left five degrees to heading ten"
[OUTPUT] (turn :heading (heading :left 5 :to 10) :agent you)

sentence: "turn left to heading three"
[OUTPUT] (turn :heading (heading :left unknown :to 3) :agent you)

sentence: "turn three"
[OUTPUT] (turn :heading (heading :to 3) :agent you)

sentence: "turn to heading three"
[OUTPUT] (turn :heading (heading :to 3) :agent you)

sentence: "you are following a continental"
[OUTPUT] (behind :goal (continental) :agent you)
```

sentence: "you are going to encounter a cessna"
[OUTPUT] (expect :plane (cessna) :agent you)

sentence: "you are going to see an alaska behind a boeing"
[OUTPUT] (expect :plane (alaska :behind boeing) :agent you)

sentence: "you are going to see an alaska"
[OUTPUT] (expect :plane (alaska) :agent you)

sentence: "you are number five"
[OUTPUT] (set-priority :new (priority :num 5 :event unknown) :agent you)

sentence: "you are number five behind a cessna"
[OUTPUT] (set-priority :new (priority :num 5 :event unknown :behind cessna) :agent you)

sentence: "you are number five for landing"
[OUTPUT] (set-priority :new (priority :num 5 :event landing) :agent you)

sentence: "you are number five for landing behind a boeing"
[OUTPUT] (set-priority :new (priority :num 5 :event landing :behind boeing) :agent you)

sentence: "you are number five for takeoff"
[OUTPUT] (set-priority :new (priority :num 5 :event takeoff) :agent you)

sentence: "you are number five for takeoff behind a cessna"
[OUTPUT] (set-priority :new (priority :num 5 :event takeoff :behind cessna) :agent you)

sentence: "you are number five to go"
[OUTPUT] (set-priority :new (priority :num 5 :event unknown) :agent you)